# Getting Started with Passbook on iOS 6

Version 1.1

## Getting Started with Passbook Overview

Introduced in iOS 6, Passbook is the simplest way for users to store all their passes in one place. Passes allow users to scan their iPhone or iPod touch to perform actions in the physical world. Examples include:

- Boarding passes
- Event tickets
- Retail coupons
- Store cards

Passbook is time and location enabled, so passes can be configured to display on the user's Lock Screen at the appropriate time and place -- such as when the user reaches the airport or walks into a store. Each time the user wakes their device, Passbook will quickly check the relevance of passes and any that are now relevant will be displayed on the Lock Screen. The user can simply slide to view the pass for quick access.

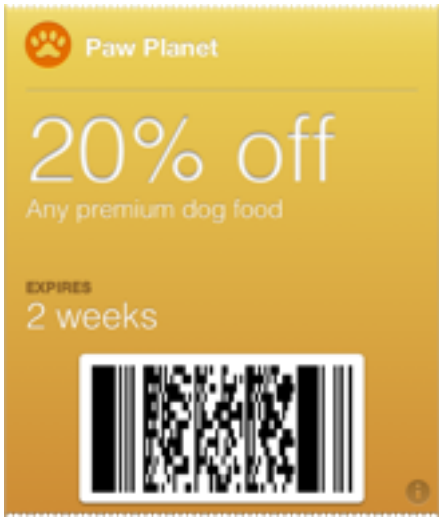
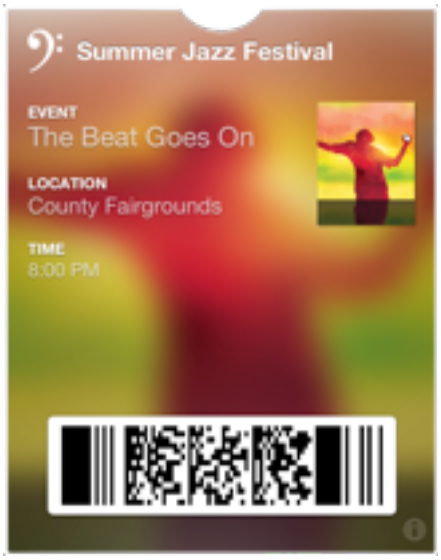
The Passbook app is where passes are stored. The front of a pass contains the most important and relevant information, typically including a barcode that can be scanned at a merchant, event, or location. Passes can be updated dynamically by way of the Apple Push Notification service which lets you easily notify the user of important updates, like when a gate changes at the airport or when the balance on a gift card changes. And passes are pushed to all of a user's devices by iCloud.

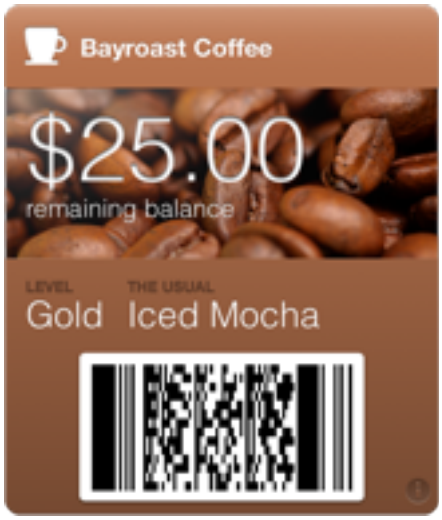
Passes can be distributed via email, the web, or directly from an app. Passes can be highly personalized with information specific to a single user or they can be generic for any user.

## Pass Styles

There are 5 different styles of passes you can create:

| Pass Style | Description | Example |
|---|---|---|
| Boarding Pass | Represents a ticket used for boarding a plane, train, boat, etc. | |

| Pass Style | Description | Example |
| --- | --- | --- |
| Coupon | Used for merchants offering a % discount or promotional item |  |
| Event Ticket | Gives access to an event at a particular time and venue |  |

| Pass Style | Description | Example |
| --- | --- | --- |
| Store Card | Can be used to represent a user's account at a store or club |  |
| Generic | Any purpose that doesn't fit in the above categories, such as a membership card | |

You must choose the correct pass style for your passes. This is important for a number of reasons:

- Users will expect certain types of information based on the visual presentation of passes. For example, event tickets have a distinctive notched edge indicating that it's a ticket to a movie or concert.
- The time window and location radius relevance is determined automatically based on the pass style. For example, the relevant time for a boarding pass is quite different from a movie ticket.

## Displaying Barcodes

Passes contain barcodes that can be scanned in order to convey information stored in the pass. Passbook supports 2D barcodes using the following formats: QR, Aztec, and PDF417. In addition to 2D barcodes, passes can display a text message below the barcode. This can be used to

display the user's membership or account number in order to allow for a human readable format, and is useful if your establishment doesn't have the ability to scan barcodes.

## Scanning Barcodes

For best results scanning barcodes presented by Passbook, use an optical scanner. Optical scanners work better for scanning from the iPhone screen than laser scanners do. Another option to consider would be to use an iOS app to use the camera on an iPhone, iPod touch or iPad to scan passes. This can provide a nice closed loop for creating, presenting, and consuming passes for your services.

Passbook will optimize the presentation of passes in order to facilitate a successful scan. For example, the screen's orientation will be locked to portrait and the backlight will temporarily be boosted to the brightest setting. Regardless of these efforts, it is imperative that you test your passes with the actual hardware you expect to use for scanning passes.

## Distributing Passes

Getting passes into Passbook is easy. You can distribute them to your users in a variety of ways:

| Distribution Means | Comments |
|---|---|
| In your app | Apps can present any type of pass to the user using a standard UIKit view controller. This gives users the option of adding the pass to Passbook. |
| Via email | Passes are digitally signed documents that can be attached to any email. When viewed on an iPhone or iPod touch with iOS 6, or on a Mac with OS X 10.8.2, users can view the pass and add it to Passbook. |
| On the web | Using Safari in iOS 6 or OS X 10.8.2, users can tap on links to passes to add them to Passbook. Distributing on the web is a convenient way to require users to sign into an account in order to receive a customized pass. |

An important point about distributing passes is that the user does not need to have your app installed on a device in order to add your pass to the Passbook library In fact, passes can exist and should be usable by users without a companion app altogether. Even if you do not have an app you can still create and distribute passes to users who can add them via email or on the web and access them directly in Passbook.

Passbook is only available on iPhone and iPod touch running iOS 6. OS X Mountain Lion (10.8.2 or later) supports adding and updating passes. Passes added in OS X will be automatically added to the Passbook library via iCloud.

### Include Your Contact Information

Passes must include your contact information, including your name, address, phone number and email address. Including this information in a separate section on the back of the pass would make it easy for users to know how to get in contact with you.

**Link to a Companion app**
On the back of the pass, you should include a link to your app which will allow a user to quickly get back to the app. If the user does not currently have the app installed, they can install it directly from the pass.

**Use the Add to Passbook badge**
The Add to Passbook badge, available for download from developer.apple.com/passbook, is recommended for use anywhere you are distributing your pass. Add it to your apps, emails and websites to give your users a branded, visual cue to tap or click to add the pass to Passbook. See the included Add to Passbook Badge Guidelines for more information.

Remember that passes can only be added to a user's Passbook from iPhone and iPod touch running iOS 6 and later, and from a Mac running OS X Mountain Lion 10.8.2 or later. The Add to Passbook badge should only be shown on these platforms. The best way to accomplish this is to pay careful attention to User-Agent strings in your implementation and only include this badge when requests are made from a Passbook-compatible platform and version.

## Updating Passes Dynamically

After a user adds a pass to Passbook, it's best to keep the pass information up to date. Some common cases would be updating the flight information for a boarding pass or updating the balance on a store card after the user makes a purchase. Updating passes can be done easily using the Apple Push Notification service (APNs). When you send a push notification to a pass, the Passbook app receives the push notification and then talks to a web-based API provided on your server to get updates to the user's pass. Users can also manually update passes by using the pull-to-refresh gesture on the back of a pass.

## Designing Great Passes

Passes have a distinct design with an emphasis on simple visual presentation on the front of the pass. Solid vibrant colors tend to work best. A solid white company logo works best. If you have a print compatible version of your logo, that might be a good option. Also keep accessibility and localization in mind when designing passes. Avoid text baked into images and instead use the various fields that are available. Text baked into images is not available to accessibility and is not localizable (unless you include localized versions of the images which can lead to passes much larger than they need to be).

The front of the pass is what users see in the Passbook app and it should be used to display the most critical pieces of information. Limiting the amount of information displayed on the front of the pass can help a user quickly get access to what they need. Passes have different types of fields that you specify in the contents of your pass. These include header, primary, secondary, and auxiliary fields. The exact layout and number of these fields displayed on the front side of passes is determined by the pass style you use.

The back side of passes can be used to display much more information, still easily available to the user. Here you can present information that the user can scroll through and allow for hyperlinks which are automatically presented as links by using Data Detectors to detect common web links or phone numbers. In addition to custom fields and arbitrary text, you can include a link to an app on the back of your passes. If the user does not have the app installed

on the device, they will see a button allowing them to install the app. If they do have the app, the button will allow the user to open the app directly from the pass.

It's easy to start off with a clean white background for a pass, but that usually ends up not looking very good. Try a bright vibrant solid color instead! If you have colors in your branding or that complement strip or thumbnail images try to highlight that color. It is very easy to try different colors so experiment with different options and you might be surprised how great a rich vibrant color can make your pass look.

## Use of Apple-issued Certificates

All passes need to be signed with an Apple-issued certificate to be recognized by Passbook. Passes should only be signed by the entity that will be distributing the pass under its own name, trademark, or brand, and that entity's information should match the information on the back of the pass. It should be obvious to users who the originator of the pass is based on the branding on the front of the pass, and that branding should be consistent with the contact information on the back of the pass.

If you want to hire a contractor to assist you in developing a pass, then you will need to add the contractor to your development team and the pass must still be signed by the entity that is distributing the pass under its name, trademark, or brand. Keep in mind that passes can only be accessed by apps developed using the same team account, and that have the appropriate entitlements. So if you intend to use a companion app in conjunction with your passes, be sure to use the same development team.

## Passbook Best Practices

Things you should do when designing and deploying passes:

- Add location and time relevance information thoughtfully. Passes will show up on the lock screen when the user wakes the device if the time and location are relevant. Making sure that the time and location are accurate will avoid cluttering the user's lock screen with passes when they aren't of use to the user.
- If you have a companion app for your pass, include a link on the back of the pass for easy access.
- Choose the most appropriate pass style for your pass.
- Use the alternate message to display meaningful text below the barcode. This can be used as an alternate to scanning, for example showing a frequent flyer number that can be entered into an airport kiosk manually.
- Be sure to handle redemption correctly. Unlike a paper coupon, users don't surrender a pass when it is scanned. It's important to have appropriate measures to handle things like duplicate scans or scans past an expiration date.
- If requiring a user to sign into a web server in order to get a pass, be sure to use https.
- Keep in mind that Passbook cannot solve the problem of duplicate passes or cases like users forwarding passes via email or capturing screen shots of passes and distributing them to other people. In order to help validate the authenticity of a pass at redemption time, you could consider pushing something to the pass that a human could verify

visually. For example, if a pass for a special deal in a retail store is only valid on a particular day, you could push a secret code word to passes that a cashier could verify.
- Push notifications should only be relevant to the passes the user has in their Passbook.
- Be aware of the iOS developer team used to register, generate and sign passes. While passes do not require an application to be usable, once installed in a user's Passbook library those passes can only be accessed by applications developed by the same team and cannot be transferred.
- Passes should be signed by the developer team that owns the name, trademark, or primary brand that is used in the pass (e.g., a boarding pass would be signed by an airline). Remember that if a sub brand is referenced in the pass (e.g., a third-party product coupon is displayed in a grocery store pass), then the primary brand holder must have any necessary rights to use that sub brand or mark.

Things you should **not** do:

- Don't overload the front of a pass. Limit the information on the front of the pass to the minimum amount of critical information, put the rest on the back of the pass.
- If nothing has changed on a pass, don't send it a push notification.
- When sending pushes using APNs, don't repeatedly open and close connections to the APNs servers. Open a connection and just leave it open.
- Don't burn text into images. Use text in the fields available instead to make accessibility and localization possible.
- Do not use the add pass view controller as a means of displaying passes in your app. The add pass view controller should only be used to let a user add a pass to Passbook. To have the Passbook app display a pass from the user's library, call the `-[UIApplication openURL:]` method and specify the return value from `-[PKPass passURL]` as the argument.

## Revocation

Apple reserves the right to revoke any Apple-issued certificate used to sign any pass at any time in its sole discretion. Apple may choose to do this for any reason, including the following:

**(a)** Any of your Apple-issued certificates have been compromised or Apple has reason to believe that they have been compromised;
**(b)** Apple has been notified or otherwise has reason to believe that your pass violates, misappropriates, or infringes the rights of a third party or of Apple (e.g., logo rights, image rights, third party data rights, rights of privacy and publicity, etc.);
**(a)** Apple has reason to believe that your pass contains false, fraudulent or misleading representations or otherwise encourages unlawful or illegal activity;
**(c)** Apple has reason to believe that your pass damages, corrupts, degrades, destroys or otherwise adversely affects the devices it operates on, or any other software, firmware, hardware, data, systems, or networks accessed or used by the pass or any app associated with the pass;
**(d)** You breach any term or condition of the iOS Developer Program License Agreement or the Registered Apple Developer terms and conditions;
**(e)** Any information or documents provided by you to Apple for the purpose of verifying your identity or obtaining any Apple-issued certificates is false or inaccurate;

**(f)** Apple is required by law, regulation or other governmental or court order to take such action;
**(g)** You request that Apple take such action;
**(h)** You misuse or overburden any services provided by Apple for passes, e.g., inappropriate use of push notifications in your pass; or
**(i)** Apple has reason to believe that such action is prudent or necessary.

*Nothing herein is intended to modify the iOS Developer Program License Agreement, Mac Developer Program License Agreement, the iOS Developer Program Enterprise License Agreement, the iOS Developer Program University Agreement, the iOS Developer Program University Student License Agreement ("Agreement") and/or the App Store Review Guidelines, as they may be modified by Apple from time to time. In the event of any conflict or inconsistency between the Agreement or Guidelines and this document, the Agreement or Guidelines shall govern. Apple may at any time, and from time to time, with or without prior notice to You modify this document as well as any features, functionality or services described herein. You understand that any such modifications may require You to change or update Your Applications at Your own cost. Apple shall not be liable for any losses, damages or costs of any kind incurred by You or any other party arising out of or related to any modification or discontinuation of this document or any of the features, functionality or services described here.*

Copyright © 2012 Apple Inc. All rights reserved.